

METHOD FOR FILTERING THE NOISE OF A DIGITAL IMAGE SEQUENCE

BACKGROUND OF THE INVENTION

Field of the Invention

The present invention concerns the processing of digital images and,
5 more particularly, a method for filtering noise in a digital image sequence.

Description of the Related Art

Digital images are currently being used in numerous applications, including those related with such traditional acquisition devices as still and video cameras. It is to be expected that ever greater use of digital images will be made
10 in such new generation devices as mobile multimedia communication terminals.

There exist numerous devices or applications that use digital images in sequence, that is to say, images acquired one after the other, separated by a brief interval of time and representing approximately the same real scene.

The speed with which the sequence is acquired, *i.e.*, the number of
15 images acquired in a given time interval, may vary according to the specific application; for example, this number is very large in digital video cameras (about 25 images per second) and smaller (about 15 images per second) in mobile communication terminals, which acquire the digital images and then transmit them in real time to a remote terminal.

20 It is well known that digital image acquisition devices, especially when they include CMOS sensors, will intrinsically introduce noise into the acquired images.

In digital image sequences noise not only degrades the quality of the images, but also reduces the encoding/compression efficiency. Indeed, the
25 acquired image sequences have commonly to be encoded/compressed by means of encoding/compression techniques that operate in accordance with, for example,

the MPEG standard or the H263 standard and nowadays are very widely used in the greater part of devices in the market today.

The encoding/compression efficiency becomes reduced by the presence of noise, because the introduced noise is typically in the form of random
5 fluctuations that reduce redundancy both within an image and between images that are temporally close to each other.

There exist numerous filtering techniques intended to reduce or eliminate the noise present in an image sequence.

Numerous attempts have been made to develop efficient techniques
10 for reducing the noise of a sequence by using various specific types of filters. Known digital filters include, for example, low-pass filters, median filters, adaptive spatial filters and recursive temporal filters with or without motion compensation.

Other prior art techniques seek to improve noise reduction efficiency in image sequences by having recourse to hybrid methods that combine digital
15 spatial filtering with digital temporal filtering.

Though the known techniques for reducing noise in image sequences are satisfactory in many respects, they are also associated with numerous drawbacks and problems that are bound up with, for example, inadequate performance, processing complexity and excessive processing costs
20 that make it difficult to employ them in portable acquisition devices of a commercial type.

BRIEF SUMMARY OF THE INVENTION

The present invention therefore sets out to make available a method for reducing noise in an image sequence. This aim is attained with a method for
25 filtering a sequence of digital images in CFA format as described in Claims 1 to 12 attached hereto.

Another object of the present invention is to provide a filter as described in Claim 13 and an acquisition device as described in Claim 14.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

Further characteristics of the invention and the advantages associated therewith will be more readily understood from the detailed description about to be given of a preferred embodiment thereof, which is to be considered as
5 an example and not limitative in any way, said description making reference to the attached drawings of which:

Figure 1 shows the block diagram illustrating a possible acquisition device that implements a method in accordance with the present invention;

Figure 2 shows the pattern of the filtering elements of a Bayer sensor
10 that can be used in the device of Figure 1;

Figure 3 schematically illustrates the succession of phases of a method in accordance with the present invention;

Figure 4 shows a selection mask for selecting green-colored pixels that can be employed in the method in accordance with the present invention;

Figure 5 shows two selection masks for selecting red-colored pixels
15 and blue-colored pixels that can be employed in the method in accordance with the present invention;

Figure 6 shows one of the phases of the succession illustrated by Figure 3 in greater detail;

Figure 7 shows an example of selecting pixels in accordance with a
20 DRT selection;

Figure 8 is a graph that illustrates the experimental results; and

Figure 9 is a schematic illustration of two filtering architectures, the first in accordance with the present invention, the second of a conventional type.

25 DETAILED DESCRIPTION OF THE INVENTION

The preferred embodiment example of the present invention relates to a portable device capable of acquiring digital image sequences for video

applications and, more particularly, concerns the noise filtering of an image sequence acquired with a digital video camera.

In this connection it should be noted that the teachings of the present invention can be extended also to applications other than those to which explicit
5 reference is made in the description about to be given, for example, to the acquisition of image sequences in mobile multimedia communication terminals of the new generation.

Figure 1 provides a very schematic illustration of a digital video camera 1 in the form of function blocks. The video camera 1 includes an
10 acquisition block 2 that comprises an optical sensor 3.

The optical sensor 3, which may be – for example - of the CCD (Charge Coupled Device) or the CMOS (Complementary Metal Oxide Semiconductor) type, is an integrated circuit comprising a matrix of photosensitive cells, each of which serves to generate an electrical signal proportional to the
15 quantity of light that strikes it during the acquisition interval. Each photosensitive cell of the sensor, which is commonly referred to by the term pixel, corresponds to a respective pixel of the digital image.

In a preferred embodiment the sensor 3 comprises an optical CFA (Color Filter Array) filter, for example, with a Bayer-type matrix.

20 As is well known to persons skilled in the art, in a sensor with a CFA filter only a single photosensitive cell is available for acquiring a pixel. The sensor is covered by an optical filter constituted by a matrix (a Bayer matrix, for example) of filtering elements, each of which is associated with a photosensitive cell. Each filtering element transmits to the photosensitive cell associated with it the light
25 radiation corresponding to the wavelength of only red light, only green light or only blue light, so that for each pixel it detects only one component (of which it absorbs no more than a minimal part). The pattern of the filtering elements in a Bayer filter is shown in Figure 2, where the letters R,G,B indicate, respectively, the red, green and blue elements.

The video camera 1 also includes an analog/digital (A/D) conversion block, indicated by the reference number 4, to translate the generated electric signal into a digital value with a predetermined number of bits (generally 8, 10 or 12 bits). One may assume, solely by way of example and without thereby
5 introducing any limitation whatsoever, that in the present invention the A/D converter 4 is such as to encode the incoming analog signals with eight-bit digital values.

On the output side of the A/D block 4 the digital image is in a video format, for example, it may be in a CFA (Color Filter Array) format, since each pixel
10 is constituted by just a single chromatic component (R, G or B). For this reason, a single one-byte digital value is associated with each pixel. In one embodiment, the digital image may be in the CFA format, but in other embodiments, other formats may be used and these are included within the concept of the invention. Thus, the reference to CFA herein should be understood to be one example of how to carry
15 out the invention.

A filtering block 5 – in this example of the Bayer type – is such as to filter the noise by operating directly on the digital CFA images of the sequence, producing for each noisy CFA image on its input side a CFA image with reduced noise on its output side.

20 A pre-processing (PrePro) block 6, active before and during the entire acquisition phase, is such as to interact with the acquisition block 2 and to extract from the CFA image a number of parameters useful for carrying out automatic control functions: self-focusing, automatic exposure, correction of sensor defects and white balancing.

25 A block 7, the IGP (Image Generation Pipeline) block, is designed to perform a processing phase that, starting from the digital CFA image, will produce a complete digital image – YCrCb format, for example – in which each pixel will have associated with it three digital values (*i.e.*, a total of 24 bits) corresponding to a luminance component Y and two chrominance components Cr and Cb. This

transformation, known by the name of color interpolation, involves a passage from a representation of the image in a single plane (Bayer plane), which nevertheless contains information relating to different chromatic components, to a representation in three planes.

5 In digital still cameras the IGP block is commonly realized in the form of a dedicated processor. In one embodiment, this is a CFA processor, which may be implemented in VLSI (Very Large Scale Integration) technology.

 Preferably, the IGP block 7 in this example is also such as to perform, over and above the interpolation, various other functions, including – for
10 example – the application of special effects, gamma correction, scaling, stabilization and other functions that will generally vary from one producer to another.

 This is followed by a compression/encoding block 8, which in this example is of the MPEG type (but could also be of other types, H263 for example),
15 and a memory unit 9.

 When shooting a video sequence with the video camera 1, the sequence images are acquired consecutively by means of the acquisition block 2, preferably separated only by a brief time interval between one image and the next. The MPEG-4 standard, for example, requires fifteen images to be acquired per
20 second.

 Hereinafter we shall use $img_1, img_2, img_3, \dots, img_{n-1}, img_n, img_{n+1}, \dots$ to indicate the images acquired in sequence: img_1 represents the first image of the sequence to be acquired, img_2 represents the second image, and so on.

 Following acquisition, each image is processed by the subsequent
25 blocks, so that in all the subsequent processing phases the images will still be processed in the temporal order in which they were acquired.

 Once they have been acquired, the sequence images are converted into digital values by the A/D converter 4.

The CFA format digital images are then sent as input to the noise filter block 5 (CFA NF) to be processed in accordance with the noise filtering method of the present invention. As output the filter block produces a sequence of filtered CFA images, respectively, $f_img_1, f_img_2, f_img_3, \dots, f_img_{n-1}, f_img_n,$
5 f_img_{n+1} , each of which has less noise than on the input side.

The filtered CFA images are then processed by the pre-processing block 6.

On leaving the pre-processing block 6, each CFA image is sent to the IGP block 7. In this block the images are subjected to a color interpolation
10 phase and therefore transformed into complete images, for example, in YCrCb format.

The color interpolation phase may be performed, among others, by means of methods that are known to a person skilled in the art and are therefore obvious from the previous description.

15 Thereafter the images are sent to the MPEG encoder block 8, which produces as its output a sequence or stream of images encoded/compressed in accordance with an MPEG encoding.

The MPEG stream of compressed images may be recorded in a memory unit 9 or sent to an external peripheral device not shown in the figure.

20 In a preferred embodiment the processing method filters the sequence of CFA images one at a time, in this example by means of a Bayer filter.

The CFA images are filtered pixel by pixel, the scanning order being such that the pixels are systematically scanned from left to right and from top to bottom. In particular, for each pixel $p_n(x,y)$ of an image img_n there is calculated a
25 respective filtered homologous pixel $f_p_n(x,y)$ of a corresponding filtered image f_img_n .

When processing the pixel $p_n(x,y)$ of the image img_n and calculating the filtered pixel $f_p_n(x,y)$, the method of the present invention makes advantageous use also of the filtered pixels forming part of an image f_img_{n-1} of

the previously filtered sequence. More particularly, it utilizes the image f_img_{n-1} obtained by filtering the image img_{n-1} that in the sequence temporally precedes the image to be filtered img_n .

Typically, therefore, three image buffers will be sufficient for carrying
5 out the filtering process: two input buffers that contain, respectively, the image img_n to be filtered (current image) and the previously filtered image f_img_{n-1} , as well as an output buffer containing the filtered current image f_img_n .

Figure 3 shows a schematic representation of the succession of
phases of a processing method 20 for reducing noise in accordance with the
10 present invention.

In particular, Figure 3 shows the phases by means of which, starting from the pixel $p_n(x,y)$ of the image img_n , there is obtained the respective homologous filtered pixel $f_p_n(x,y)$ of the corresponding filtered image f_img_n .

Given the input pixel $p_n(x,y)$ to be filtered, a first selection phase 21
15 (SW_sel) selects a first pixel set $SW_n(x,y)$ comprising the said pixel $p_n(x,y)$ and a plurality of pixels forming part of the image img_n in the neighborhood of said pixel. In one embodiment, the phase selects adjacent pixels that have associated with them the same color (R, G or B) as the pixel to be filtered.

In a preferred embodiment, the selection is performed by using
20 selection masks (or matrices) SM_G , SM_R , SM_B that differ on the basis of the color of the pixel $p_n(x,y)$ to be filtered, but all of dimension 5×5 , for example, like those shown in Figures 4 and 5.

Figure 4 shows a selection mask SM_G for the green pixels (G) in accordance with a particular embodiment of the invention. In the selection phase
25 21 the mask SM_G is aligned with the image to be filtered in such a way that that G_0 corresponds to the green pixel $p_n(x,y)$ to be filtered. In this way the mask SM_G will select a first set of pixels $SW_n(x,y)$ comprising the green pixel $p_n(x,y)$ corresponding to G_0 and eight adjacent pixels situated, respectively, in positions

corresponding to the pixels G_1, \dots, G_8 of the mask SM_G shown in the figure. This set defines a spatial working window $SW_n(x,y)$ for the green pixel to be filtered.

Analogously, Figure 5 shows the selection masks SM_R and SM_B to be used, respectively, when the pixels to be filtered are red or blue.

5 The mask SM_R for the red pixels is such as to select a first set of pixels $SW_n(x,y)$ comprising the red pixel $p_n(x,y)$ to be filtered corresponding to R_0 and eight adjacent red pixels situated, respectively, in positions corresponding to the pixels R_1, \dots, R_8 of the mask.

 It should be noted that in this particular embodiment the selection
10 mask SM_B for the blue pixels is identical with the selection mask SM_R for the red pixels. This choice is possible thanks to the particular pattern in which the filtering elements are arranged in a Bayer-type sensor.

 This brings with it the advantage that the selection phase 21
(SW_sel) has to discriminate only between two possible cases, namely to
15 distinguish whether the pixel $p_n(x,y)$ to be filtered is or is not green.

 Coming back to Figure 3, a second selection phase 22 (TW_sel) selects a second set of pixels $TW_n(x,y)$, comprising pixels forming part of the previously filtered image f_Img_{n-1} and arranged in corresponding positions, *i.e.*, homologous with the pixels of the first $TW_n(x,y)$.

20 To this end it will be advantageous to use the selection matrices described above, but this time applied to the previously filtered image f_Img_{n-1} .

 The pixel set obtained in this manner defines a temporal working window $TW_n(x,y)$ for the pixel to be filtered.

 The temporal and spatial working windows represent the set of pixels
25 that will play a part in the subsequent phases of the filtering process of the pixel $p_n(x,y)$.

 As is well known to a person skilled in the art, when digital image sequences are filtered, a filtered pixel can be obtained by appropriately combining

a certain number of pixels that are adjacent to it either in space (spatial filtering), in time (temporal filtering) or in space/time (spatio-temporal filtering).

In particular, as will be described in greater detail later on, the method of the present invention decides pixel by pixel whether the filtering to be used is to be exclusively spatial or, on the other hand, spatio-temporal. Advantageously, the decision regarding the type of filtering to be employed will be bound up with the amount of motion between successive images of the sequence, since this will make it possible to avoid motion compensation, a computationally very costly operation.

When spatial filtering is employed, the operation will involve only the pixels of the spatial working window $SW_n(x,y)$, otherwise use will be made of the pixels forming part of both the windows.

Once the two working windows – respectively in space and time – have been obtained, a first noise estimation phase 23 (Snoise_est) has as its first step the making of an estimate of a statistical parameter $NL_n(x,y)$ representative of the noise level present on the pixel $p_n(x,y)$ and the respective spatial working window $SW_n(x,y)$. Henceforth we shall refer to this noise as spatial noise, while the phase will be referred to as spatial noise estimation.

In greater detail, the first step of the estimating phase 23 (Snoise_est) is to make a preliminary noise estimate (*i.e.*, to estimate a statistical parameter representative of the noise level) by means of a local calculation, that is to say, calculated for the pixels of the spatial working window $SW_n(x,y)$. As second step it obtains the definitive spatial estimate $NL_n(x,y)$ by modifying the preliminary estimate on the basis of a spatial estimate of the noise specific for the color of the pixel to be filtered and specific also for the image Img_n .

In greater detail, again, the spatial noise estimate is obtained by means of a computation of the recursive type that is made by taking into account not only a preliminary and local noise calculation, but also the spatial estimate of

the noise level made for the last filtered pixel of the image Img_n having the same color as the pixel to be filtered $p_n(x,y)$.

Stated in mathematical terms, in the case in which, for example, $p_n(x,y)$ is a green pixel, we have:

$$5 \quad NL_n(x,y) = NL_n^G(x,y) = k_n(x,y) \times N[SW_n(x,y)] + (1 - k_n(x,y)) \times NL_n^G(pp^G) \quad (1)$$

where the superscript "G" indicates that the term relates to the color green,

$N[SW_n(x,y)]$ is the preliminary estimate calculated for the spatial working window $SW_n(x,y)$, $k_n(x,y)$ is a multiplication factor comprised between zero and one and determines the strength of the spatial filter, $NL_n^G(pp^G)$ is the spatial noise estimate

10 made for the green pixel pp^G of the image Img_n that immediately precedes the green pixel to be filtered $p_n(x,y)$ in the order in which the image Img_n is scanned.

Obviously, if the pixel to be filtered $p_n(x,y)$ is the first pixel of the respective color to be filtered in the image Img_n , only the preliminary estimate $N[SW_n(x,y)]$ will be available. In that case we can either put, for example,

15 $k_n(x,y)=1$ solely for the pixel $p_n(x,y)$ or, alternatively, assign an arbitrary and preferably small value to the quantity $NL_n^G(pp^G)$.

The first phase of the spatial noise estimation 23 (Snoise_est) may be carried out, for example, as described in detail in European Patent Application No. 01830562.3 filed in the name of the present applicant, which is to be deemed

20 to be wholly incorporated herein by reference. The meaning of the quantities $NL_n(x,y)$, $N[SW_n(x,y)]$, $k_n(x,y)$, $NL_n^G(pp^G)$ and the manner in which they are calculated are likewise explained in that document. In this connection please refer to formulas (1), (2), (3), (4), (5), (6), (7), (8) and Figures 7, 8A, 8B, 10 (and the descriptions relating thereto) of the aforesaid patent application No. 01830562.3.

25 For further details of the spatial noise estimation as described hereinabove, especially as expressed in equation (1), reference should also be made to US Patent 6,108,455.

The spatial noise estimate $NL_n(x,y)$ as calculated in this manner is used for regulating the degree or strength of the filtering in the case in which the filtering of the pixel $p_n(x,y)$ is exclusively of the spatial type.

Once the spatial noise estimation phase of $Snoise_est$ 23 has been completed, a subsequent inhomogeneity estimation phase $Text_est$ associates an inhomogeneity index (or “texture” degree) $T_D(x,y)$ with the pixel to be filtered $p_n(x,y)$ on the basis of a measure of the inhomogeneity (or, analogously, of the homogeneity) of the pixels forming part of the spatial working window $SW_n(x,y)$.

The inhomogeneity index $T_D(x,y)$ serves to decide whether the pixel $p_n(x,y)$ does or does not form part of a homogeneous region, this with a view to establishing whether or not the pixel in question (and therefore the corresponding spatial working window) will have to contribute to a spatio-temporal noise estimate to be described in greater detail further on.

In fact, a homogeneous region can provide reliable information about the effective noise present in the image, because the fluctuations (*i.e.*, the differences) between pixels forming part of a homogeneous region are substantially to be attributed to random noise.

Persons skilled in the art are familiar with different metrics for calculating an inhomogeneity measure associated with a set of pixels, and for this reason we shall not here delve further into this matter. Among these metrics we shall here cite the following solely by way of example: maximum difference, minimum difference, MAD (Mean of the Absolute Differences), standard deviation, extraction of a distribution parameter from a histogram of the digital values of the pixels.

In an embodiment that is particularly advantageous from a computational point of view, the parameter $k_n(x,y)$ used in the spatial noise estimation phase 23 ($Snoise_est$) is calculated on the basis of an inhomogeneity/homogeneity measure. More particularly, it is obtained on the basis of a calculation of the differences between the pixel to be filtered and the

other pixels of the spatial working window. In that case the inhomogeneity estimation phase 24 (Text_est) can be inserted in the spatial noise estimation phase 23 (Snoise_est) by calculating a single inhomogeneity measure that can be used both for estimating the spatial noise and for associating an inhomogeneity index $T_D(x,y)$ with the pixel to be filtered.

A comparison phase 25 is used to verify whether the inhomogeneity index $T_D(x,y)$ of the pixel $p_n(x,y)$ is smaller than a predetermined threshold value T_h (*i.e.*, whether the pixel forms part of a region deemed to be homogeneous). Namely, when the inhomogeneity index T_D is smaller than a threshold T_h , there is a local noise estimation phase 26 before the motion detection phase 27, but if T_D is larger than T_h , there is a motion detection phase 27 immediately after the comparison phase.

If this is not the case, the next step is the motion detection phase 27 (Mot_det). But if the pixel forms part of homogeneous region, the motion detection phase 27 is preceded by a second local noise estimation phase 26 (L_STnoise_est), *i.e.*, performed on the pixel to be filtered, that serves to obtain a global noise estimate (*i.e.*, for the entire image that is being processed).

The local noise estimation phase 26 (L_STnoise_est) estimates a parameter - which may be statistical, for example - representative of the noise locally present in the spatial working window. This parameter is calculated, for example, as a local standard deviation σ_{n+1}^{Loc} of the spatial working window of the pixel $p_n(x,y)$ or as some other analogous energy measure.

Given the spatial working window $SW_n(x,y)$ of the pixel $p_n(x,y)$, the local standard deviation σ_{n+1}^{Loc} can be calculated in accordance with the following formula:

$$\sigma_{n+1}^{Loc}(x,y) = \sqrt{\frac{1}{N-1} \sum_{(x,y) \in SW_n(x,y)} (p_n(x,y) - m)^2} \quad (2)$$

where N is the number of pixels forming part of the spatial working window $SW_n(x,y)$ (in this case $N=9$) and m is the mean of the digital values of these pixels.

As already mentioned (and as is to be explained in greater detail further on), once the filtering of the image Img_n has been terminated, the various
5 local standard deviations σ_{n+1}^{Loc} calculated in this manner for the pixels of the image Img_n that are deemed to form part of homogeneous regions will be used for updating a global noise estimate σ_{n+1}^{GL} , which we shall hereinafter refer to also as spatio-temporal noise estimate. In particular, this global estimate will be used for the spatio-temporal filtering of the subsequent image Img_{n+1} .

10 The movement detection phase 27 (Mot_det) compares the pixels of the temporal working window $TW_n(x,y)$ with the pixels of the spatial working window $SW_n(x,y)$ in order to ascertain the presence of motion between the two working windows and possibly evaluate its magnitude.

As is well known to a person skilled in the art, when using a “non-
15 compensated motion” approach, appropriate precautions have to be taken to avoid the introduction during the spatio/temporal filtering of artifacts due to the motion between consecutive images. In particular, care must be taken to assure that the two working windows will not contain incongruent data on account of the motion between consecutive images or parts of them.

20 For example, it may happen that one working window contains pixels that form part of an object, while the other window contains pixels that form part of the background, because the object has moved between one image and the next.

Typical examples of artifacts that could be produced in these cases are the presence of troublesome trails and so-called “ghost images” that become
25 visible in the filtered image due to residual information of previous images.

In a preferred embodiment, the motion detection phase (Mot_det) calculates as measure of motion a measure $M(x,y)$ that is the sum of the absolute differences (SAD) between the pixels of the temporal working window and the pixels of the spatial working window. The greater the differences between the two

windows, the greater will be the value of this measure $M(x,y)$, which can therefore be representative of the motion between the two windows.

The standard SAD measure as an isolated item is well known to persons skilled in the art and thus need not be described in detail herein.

5 In a particularly advantageous embodiment variant when used with the present invention, the motion measure $M(x,y)$ is a “modified” SAD. This measure is calculated by determining the difference in absolute value between the two working windows pixel by pixel, thus obtaining a working window difference $DW_n(x,y)$ given by:

10
$$DW_n(x,y) = |SW_n(x,y) - TW_n(x,y)| \quad (3)$$

Subsequently one proceeds to calculate the mean W_{avg} of the pixels of the operating window difference $DW_n(x,y)$, thus obtaining the modified SAD measure $M(x,y)$, which is given by:

$$M(x,y) = SAD(DW_n(x,y) - W_{avg}).$$

15 The modified SAD measure obtained in this manner is advantageous, because it makes it possible to avoid a change in lighting conditions being erroneously interpreted as a motion.

With a view to avoiding an excessive sensitivity of the SAD value to the digital values of the pixels of the two working windows (which are contaminated
20 by noise), another particularly advantageous embodiment variant makes it possible to introduce a slight quantization of the pixel values by reducing the accuracy of the pixel values from eight to seven bits when calculating the SAD.

Another comparison phase 28 then checks whether the motion measure $M(x,y)$ of the pixel $p_n(x,y)$ is greater than a predetermined threshold value
25 M_h .

When this is the case, the system concludes that there is excessive change between the two working windows and the subsequent filtering phase 29 (S_filter) is therefore exclusively of the spatial type.

The exclusively spatial filtering produces the filtered pixel $f_{p_n}(x,y)$ from the pixels of the spatial working window. The strength of the filtering is regulated by the estimate of the spatial noise level $NL_n(x,y)$ calculated in noise estimation phase 23 (Snoise_est). In a preferred embodiment, the spatial digital filtering is carried out in accordance with the technique described in the previously mentioned European Patent Application No. 01830562.3, which obtains the filtered pixel as a weighted average (mean) of the pixels of the spatial working window (see, in particular, formula (9) of said application).

Obviously, the exclusively spatial filtering is also carried out for all the pixels of the first image of the sequence, because temporal data are not yet available in this case.

On the other hand, when the motion measure $M(x,y)$ is smaller than the predetermined threshold value M_h , the subsequent filtering phase 30 (ST_filter) is of the spatio/temporal type and is illustrated in greater detail in Figure 6.

ST_filter 30 consists of a first filtering phase 33 (Duncan_Filt), which produces a provisional filtered pixel $d_{p_n}(x,y)$ in accordance with a filtering technique that is known by the name of Duncan filtering and will be described in greater detail further on. In this phase the provisional filtered pixel $d_{p_n}(x,y)$ is obtained from a subset of pixels forming part of both the working windows.

As can be seen in Figure 6, when the detected motion is deemed to be sufficiently small, *i.e.*, smaller than a further predetermined threshold value M_l that is smaller than the threshold value M_h , the provisional filtered pixel is not subjected to any further processing and one simply puts:

$$f_{p_n}(x,y) = d_{p_n}(x,y) \quad (4)$$

In this case, therefore, the pixel $p_n(x,y)$ is effectively filtered by means of a Duncan spatio/temporal filtering method.

Vice versa, *i.e.*, in the case in which the detected motion is not negligible, the provisional pixel $d_p_n(x,y)$ is subjected to a further processing phase
5 35 (Smooth_Filt), which produces the “definitive” filtered pixel $f_p_n(x,y)$ in accordance with a smoothing operation as defined by the following formula:

$$f_p_n(x,y) = \beta_n \times d_p_n(x,y) + (1 - \beta_n) \times p_n(x,y) \quad (5)$$

where β_n is a multiplying factor comprised between 0 and 1 that may either depend on the motion measure $M(x,y)$ or may be the same for all the images of the
10 sequence. In a preferred embodiment, for example, β_n is equal to about 0.75.

Following the smoothing operation, the definitive filtered pixel is obtained from a portion (in this example 75%) of the value provided by the Duncan filtering and a portion (25%) of the value of the unfiltered pixel. In other words, the definitive filtered pixel is obtained from the sum of a fraction of the provisional
15 filtered pixel and a fraction of the unfiltered pixel.

This is done because, whenever there are non-negligible incongruencies due to motion between the working windows, it is important that one should be able to “neglect” the temporal information to a somewhat greater extent and attribute a little more importance to the current image that is being
20 filtered.

We shall now describe a particularly advantageous embodiment of the Duncan filtering phase.

Duncan filtering first selects a subset or “range” of pixels forming part of the two working windows by means of the so-called “Duncan Range Test” or
25 “DRT” (in this connection see “Multiple range and multiple f-tests”, D.B. Duncan, Biometrics, vol. 11., pp. 1-42, 1955).

The application the DRT to digital filtering is known, for example, from European Patent Application EP 1 100 260 A1 by the present applicant.

The subsequent filtering operations of the pixel $p_n(x,y)$ are then performed only on the pixels that form part of the range selected by means of the DRT.

The purpose of the selection effected by means of the DRT is to
5 exclude any pixels that, though forming part of the working windows, have had their value corrupted to an excessive extent by noise.

For example, such pixels may be present due to the effect of a particular noise – known as “salt and pepper noise” – capable of bringing the digital values of some pixels up to the maximum value or down to the minimum
10 value of the scale of the possible digital values.

Selection by means of the DRT is also intended to exclude any pixels that are very different from the pixel to be filtered $p_n(x,y)$, for example, on account of a different information content. One may think, for example, of the case in which the pixel to be filtered $p_n(x,y)$ forms part of an “edge”, while the working
15 windows contain pixels that form part of the background of the scene. If the background pixels were not excluded, the image would suffer a considerable loss of definition as a result of the filtering.

A selection made by means of the DRT has to identify a digital value interval SI (selection interval) having an appropriate width S such as to contain the
20 largest possible number of pixels (in this case forming part of the working windows) similar to the pixel to be filtered $p_n(x,y)$. The width S of the selection interval SI is correlated with the standard deviation of the noise to be filtered, which is assumed to be known.

It is not necessary for this interval to be centered around the pixel to
25 be filtered: if this were to be the case and if the pixel to be filtered were, for example, highly corrupted by noise, the test would exclude pixels useful for the filtering.

In a preferred embodiment, the noise standard deviation used for the DRT selection during the filtering of the pixel $p_n(x,y)$ of the image Img_n is the global

spatio/temporal noise estimate σ_n^{GL} calculated in the filtering of the previous image Img_{n-1} . This choice is particularly advantageous from the point of view of computational resource optimization: in this way, in fact, one avoids having to carry out a complete scanning of the image Img_n that is to be filtered (an operation that
5 serves only to estimate the noise) prior to the filtering in the proper sense of the term.

The theory of DRT selection provides all the instruments needed for calculating the width S from the standard deviation of the noise and determining the optimal selection interval SI . The implementation of these instruments,
10 however, is very costly in computational terms and cannot readily be conciliated with the stringent requirements imposed by real-time image processing.

In a preferred embodiment, an optimal compromise between reliability of the result and computational complexity is obtained by performing the selection of the pixel subset with one of the following three intervals $SI1$, $SI2$, $SI3$,
15 as shown in Figure 7, where:

- the interval $SI1$ of width S is centered around the digital value DV of the pixel to be filtered $p_n(x,y)$;
- the interval $SI2$ of width S is centered around the digital value $DV = p_n(x,y) + \sigma_n^{GL}$;
- 20 • the interval $SI3$ of width S is centered around the digital value $DV = p_n(x,y) - \sigma_n^{GL}$.

The interval to be chosen from among these three intervals $SI1, SI2, SI3$ is the one that contains the largest number of pixels, which in Figure 7 is the interval $SI1$. In this way good results are obtained even when the pixel to be
25 filtered $p_n(x,y)$ is a very noisy pixel.

Still in a preferred embodiment, moreover, the width S of the selection interval SI is calculated as:

$$S = 3 \times \sigma_n^{GL} \quad (6)$$

In a variant that is computationally costlier but yields optimized performances, the choice of the selection interval is made by using appropriate weighting functions in accordance with the method described in the aforementioned European Patent Application EP 1 100 260 A1 (where particular
5 reference should be made to Figures 1b and 4).

Once the pixels P_j most similar to the pixel to be filtered and contained in the two working windows and the selection interval SI have been identified by means of the DRT, the provisional filtered pixel $d_p_n(x,y)$ is calculated as the weighted mean of these pixels or, put in mathematical terms:

$$10 \quad d_p_n(x,y) = \sum_{P_j \in SI} a_j P_j \quad (7)$$

where, preferably, the weighting coefficients a_j are calculated as in the aforementioned European Patent Application EP 1 100 260 A1 (where particular reference should be made to page 6, lines 41-50).

Coming back to Figure 3, the filtering – which, as previously
15 explained, takes place in accordance with either phase 29 (S_filter) or phase 30 (ST_filter) – is followed by a control phase 31 that checks whether the pixel $p_n(x,y)$ that has just been filtered is the last pixel of the image Img_n . If $p_n(x,y)$ is not the last pixel, the method represented as a succession of phases in Figure 3 is applied to the next pixel in the scanning order, for example, the pixel $p_n(x,y+1)$.

20 When it is the last pixel, on the other hand, there follows a global noise estimation phase of the spatio-temporal type based on the numerous local estimates σ_{n+1}^{Loc} calculated for the pixels of the image Img_n deemed to form part of homogeneous regions during the local noise estimation phase 26. In particular, these estimates are used to update a global noise estimate σ_{n+1}^{GL} of the spatio-
25 temporal type that will subsequently be used in the spatio-temporal filtering of the next image Img_{n+1} . Preferably, the global estimate σ_{n+1}^{GL} should be representative

of the standard deviation of the noise and be calculated as the mean of the numerous local estimates (standard deviations) σ_{n+1}^{Loc} .

In some situations it may however happen that adjacent images have excessively discordant values of the global standard deviation σ_{n+1}^{GL} and the spatio-
5 temporal filtering could therefore filter adjacent images with intensities that are too widely different. This would give rise to a bothersome flickering in the reproduction of the sequence.

With a view to avoiding this drawback, a preferred embodiment modifies the global standard deviation σ_{n+1}^{GL} , originally calculated as the mean of
10 the local standard deviations, by obtaining a time average in a recursive manner of a certain number (for example: two) of global standard deviations σ^{GL} relating to consecutive images. Put in mathematical terms, we thus have:

$$\sigma_{n+1}^{GL} = \gamma \times \sigma_{n+1}^{GL} + (1 - \gamma) \times \sigma_n^{GL} \quad (8)$$

where γ is a number comprised between 0 and 1, σ_n^{GL} is the global noise estimate
15 as updated during the filtering of the previous image Img_{n-1} and used in the spatio-temporal filtering of the current image Img_n . For example, the number γ may be equal to about 0.75.

We shall now describe some embodiments alternative to the particular method described hereinabove by reference to Figure 3.

20 As compared with Figure 3, the motion detection block 27 (Mot_det) of one of these embodiment variants limits itself to detecting the presence/absence of motion and as output provides a binary measure $M(x,y)$ indicative of the presence/absence of motion. For example, the output may be $M(x,y)=1$ when the presence of motion is detected, otherwise the output will be $M(x,y)=0$. In this case
25 the value of the threshold M_h may be chosen, for example, as equal to 0.5.

Phase 24 (Mot_det) may detect the presence of motion by means of a "trail detection" method that operates by simply calculating the differences the two working windows pixel by pixel. If these differences all have the same sign, be

it positive or negative, the system detects the presence of motion, otherwise it detects the absence of motion.

When the presence of motion is detected, the next step is the previously described spatial filtering phase 29.

5 In the contrary case, *i.e.*, when no motion is detected, this is followed by a spatio-temporal filtering phase 30 (ST_filter) in which the filtered pixel is obtained by means of a Duncan filtering phase 33 (Duncan_Filt) – see Figure 6 - that may or may not be followed by a phase of smoothing filtering 35 (Smooth_Filt).

10 In yet another embodiment variant, the two filtering phases 29 and 30 (S_filter and ST_filter), which are, respectively, a spatial filtering and a spatio-temporal filtering, obtain the weighted means for calculating the filtered pixel $f_{pn}(x,y)$ by replacing some of the pixels that form part of the weighted mean by their respective filtered values whenever this value is already available in the
15 output buffer.

Experimental results have shown that the proposed filtering method is capable of providing concrete advantages in terms of both image quality and encoding/compression efficiency, and this notwithstanding the fact that it calls for the allocation of only modest computational and memory resources. This renders
20 use of the method of the present invention particularly advantageous in applications that call for real-time processing capacity.

Referring to Figure 8, the curve denominated “CFA_filtered” reproduces, image by image, a quality measure for a sequence of three hundred images that were filtered in accordance with the present invention. The measure
25 was obtained from filtered and interpolated images.

The second curve in Figure 8, indicated by the denomination “Noisy”, reproduces the same measure as obtained on the same sequence prior to filtering; in this case, once again, the measure was obtained from interpolated images.

The results of Figure 8 refer, in particular, to a measure known as PSNR (Peak-to-Peak Signal to Noise Ratio). The PSNR is a standard measure and is representative of the quality of an image; more particularly, it indicates the signal quantity present in an image as compared with the quantity of noise.

5 The results of Figure 8 show that the filtered sequence is characterized by a higher PSNR measure (the gain is typically of the order of 3 dB), which is indicative of a better quality.

Figure 9 shows two processing schemes that can be used for obtaining a sequence of filtered images encoded/compressed in accordance with
10 the MPEG4 standard from a sequence of noisy CFA images.

More particularly, in the first of the two procedures, here denominated E1, the CFA images are filtered by means of a method in accordance with the present invention by the filter 5 (CFA NF), after which they are interpolated by the block 7 (IGP) and encoded/compressed by the block 8 (MPEG4-Encoder).

15 In the other procedure, here denominated E2, the noisy CFA images are first interpolated by the block 7 (IGP), after which – following a conventional filtering scheme – they are filtered by means of a filtering method with motion compensation immediately prior to MPEG encoding/compression in Block 8, this method being indicated as MC_NF. In particular, the MC_NF method uses the
20 motion estimate and the motion compensation of the MPEG encoder to perform a spatio-temporal digital filtering with motion compensation.

Experimental results have shown that processing procedure E1, *i.e.*, the one in accordance with the present invention, makes it possible to obtain a gain of 20% in terms of bit rate with respect to the conventional procedure E2.

25 This can be explained by considering the fact that processing procedure E2 estimates the motion and the motion compensation from images that are affected by noise and cannot therefore furnish an optimal result, with consequent adverse effects as regards both image quality and compression efficiency.

The filtering method in accordance with the invention, which has been described hereinabove by reference to a preferred embodiment, can be implemented by utilizing hardware, software or a combination of hardware and software. In the latter case the method may be implemented in an application-specific integrated circuit (ASIC circuit).

When it is implemented in a device for acquiring image sequences, the method in accordance with the present invention can be advantageously carried out by means of processing resources (DSP, for example) shared with other applications within the said device.

Obviously, a person skilled in the art, especially when having to satisfy contingent and specific needs, could introduce numerous modifications and variants into the proposed method of filtering a digital image sequence, though without thereby overstepping the protection limits of the invention as defined by the claims set out hereinbelow.

All of the above U.S. patents, U.S. patent application publications, U.S. patent applications, foreign patents, foreign patent applications and non-patent publications referred to in this specification and/or listed in the Application Data Sheet, are incorporated herein by reference, in their entirety.

From the foregoing it will be appreciated that, although specific embodiments of the invention have been described herein for purposes of illustration, various modifications may be made without deviating from the spirit and scope of the invention. Accordingly, the invention is not limited except as by the appended claims.